DETAILED ACTION

a. This action is taken to response to remarks and amendments filed on 11/23/2009.

b. Rejections and/or objections not reiterated from previous office actions are hereby withdrawn.

c. Claims 1-8, 10-18, 20-22, 24, 26-30, 32-34, 36, and 39-43 are pending. Claims 1, 13, 22, 27, 29, 30, 32, 36, and 39 have been amended. Claims 23, 31, and 37 have been canceled. Claims 1, 13, 22, 27, 36, and 39 are independent claims.

Claim Rejections – 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 39-43 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claim 39 recites a system, however, the use of various components and elements that would be reasonably understood by one of ordinary skill in the art to mean software, software based component implementation, or an abstract concept based on software. Examples of components and concepts used in the claim are "a controller" and "a network boot service" and other such terms that are interpreted to mean abstract concepts and software implementations. Furthermore, instant specification (page 27, line 10, the process of Fig 8 is performed by controller 122 of Fig. 2, and may be implemented in software…) also provides intrinsic evidence

to be interpreted as software. Since a computer program is merely a set of instructions capable of being executed by a computer without the necessary physical articles or objects to constitute a machine or a manufacture and do not fall within a statutory category of invention.

Regarding claims 40-43 depend from rejected claim 39, comprise the same deficiencies as those claims directly or indirectly by dependence, and are therefore rejected on the same basis.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 1-4, 6-8, 10-13, 15-18, 20-21, 27-30, 32-34, and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Denby et al. (US Patent 6,976,062, hereinafter Denby) in view of Devine et al. (US Patent 6,944,662, hereinafter Devine).

As to claim 1, Denby discloses **a method, implemented in a device** (Fig. 1)**, the method comprising**:

obtaining a task sequence at the device that describes a set of one or more steps to be carried out in managing multiple additional devices (col. 7, lines 14-34, The parameter list in the upgrade.ini file controls the upgrade behavior, col. 11, line 58 to col. 13, line 15, upgrade.ini file typically contains tags and parameter definitions, the parameters will be used to select the sections of the upgrade that need to be performed, col. 2, lines 8-9, initiate simultaneous upgrades to multiple target devices);

generating a job tree at the device representing the set of one or more steps (col. 7, lines 27-45, col. 11, line 58 to col. 13, line 15, the parameters will be used to select the sections of the upgrade that need to be performed, which are the job tree being performed by upgrade)**, the set of one or more steps configured to at least one of**:

configuring firmware of the multiple additional devices (col. 2, lines 23-28, col. 2, line 66 to col. 3, line 6, upgrade the operating system, firmware, applications and data files)**;**

downloading an operating system to the multiple additional devices (col. 2, lines 16-18, install a new version of the operating system)**;**

rebooting the multiple additional devices (col. 7, lines 14-18, 24-25, The rebooting and process control is done automatically); **or**

configuring the operating system of the multiple additional devices (col. 2, lines 8-9, col. 3, lines 33-35, col. 4, lines 10-16, col. 14, line 60 to col. 15, line 3, col. 19, lines 37-39, A new config.sys and startup.cmd will be copied over the old so that when the DCS 300 reboots); **and**

sending one or more commands configured to carry out the set of one or more steps in accordance with the job tree (col. 7, lines 24-59, col. 11, line 25-26, col. 17, lines 46-56, start upgrade from the GUI by calling upgrade utility), but does not explicitly disclose **wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices**.

However, Devine discloses wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices (Fig. 6, clients connected to server, Fig. 10, clients receive data service, col. 9, lines 1-17 and 61-67, clients exchange data with servers and software upgrades are performed by distributed administrator, a server, col. 46, lines 41-49 distributed administrator provides both synchronous and asynchronous services, col. 47, lines 27-31, service is delivered synchronously or asynchronously to clients, the connected devices).

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Denby with the teachings Devine to reduce downtime and to free developer

from firmware upgrade as it has been realized that the asynchronously and synchronously software upgrade is efficient and cost-effective (col. 52, lines 39-65).

As to claim 2, Denby in view of Devine discloses the method as recited in claim 1, wherein the set of one or more steps includes steps for automatically deploying an operating system on the multiple additional device (col. 2, lines 16-18).

As to claim 3, Denby in view of Devine discloses the method as recited in claim 1, wherein carrying out the set of one or more steps comprises:

carrying out a first step of the set of one or more steps (col. 11, lines 5-13); and carrying out the remaining steps of the set of one or more steps only if the first step is completed successfully (col. 11, lines 5-13).

As to claim 4, Denby in view of Devine discloses the method as recited in claim 1, wherein carrying out the set of one or more steps causes the device to have firmware on the multiple additional device configured (col. 2, lines 23-28, col. 2, line 66 to col. 3, line 6, upgrade the operating system, firmware, applications and data files) and an operating system to be deployed on the multiple additional device (col. 2, lines 16-18, install a new version of the operating system).

As to claim 6, Denby in view of Devine discloses the method as recited in claim 1, wherein one of the steps comprises another task sequence (col. 20, lines 48-52, reboot step comprises sequence of saving and rebooting system).

As to claim 7, Denby in view of Devine discloses the method as recited in claim 1, wherein one of the steps comprises an operation to be performed (col. 7, lines 27-34).

As to claim 8, Denby in view of Devine discloses the method as recited in claim 1, wherein the job tree comprises a parent node corresponding to the job and one or more child nodes (col. 7, lines 27-34, is a tree of sequential step nodes for upgrade), wherein each child node corresponds to one of the one or more steps (col. 7, lines 27-34, where a tree of sequential step nodes upgrade in which each step is a child node of its preceding step).

As to claim 10, Denby in view of Devine discloses the method as recited in claim 1, wherein the task sequence comprises a user-defined task sequence (Fig. 2, col. 11, lines 63-64).

As to claim 11, Denby in view of Devine discloses the method as recited in claim 1, wherein the task sequence comprises a user-selected task sequence (Fig. 2, col. 6, lines 54-55, col. 11, lines 63-64).

As to claim 12, Denby in view of Devine discloses the method as recited in claim 1, further comprising recording the set of one or more steps in a log (col. 15, lines 46-63, col. 17, lines 12-15).

As to claim **13**, is directed to a computer readable medium (Col. 4, lines 3-6) carrying instructions for performing the methods of claim 1 therefore rejected along the same rationale.

As to claims **15-18 and 20-21**, are directed to a computer readable medium carrying instructions for performing the methods of claims 2-3, 6-8, 12 respectively and are therefore rejected along the same rationale.

As to claim 27, is directed to a computer readable medium (col. 4, lines 3-6) carrying instructions for performing the methods of claim 1 therefore rejected along the same rationale.

As to claim 28, has the same subject matter as of claim 10 and is rejected under the same rationale.

As to claim 29, Denby in view of Devine discloses the one or more computer readable media as recited in claim 27, wherein the ordered series of steps comprises a tree having a plurality of nodes (col. 7, lines 27-34, is a tree of sequential step nodes for upgrade), wherein each of the one or more elements for each step is represented by one of the plurality of nodes (col. 7, lines 27-34, where a tree of sequential step nodes upgrade in which each step is a child node of its preceding step).

As to claim **30**, Denby in view of Devine discloses the one or more computer readable media as recited in claim 29, wherein the ordered series of steps includes a one to one corresponding of elements to steps (col. 7, lines 27-34, where a tree of sequential step nodes for upgrade in which each step maps one to one corresponding a task being performed).

As to claims **32-34**, are directed to a computer readable medium carrying instructions for performing the methods of claims 3, 6-7 respectively and are rejected along the same rationale.

As to claim 36, is directed to a system claim carrying instructions for performing the methods of claim 1 and is rejected along the same rationale.

Claims 22, 24 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Denby et al. (US Patent 6,976,062, hereinafter Denby) in view of Paul et al. (U.S. Patent Application 2003/0005096, hereafter Paul) and further in view of Devine et al. (US Patent 6,944,662, hereinafter Devine).

As to claim 22, Denby discloses **a method, implemented in a device** (Fig. 1)**, the method comprising**:

obtaining a user-defined task sequence at the device that describes an action to be carried out to automatically deploy an operating system to multiple additional devices (col. 7, lines 14-34, The parameter list in the upgrade.ini file controls the upgrade behavior, col. 11, line 58 to col. 13, line 15, upgrade.ini file typically contains tags and parameter definitions, the parameters will be used to select the sections of the upgrade that need to be performed, col. 1, line 64-65, col. 2, lines 8-9, an automatic implemented upgrade process includes a step to initiate simultaneous upgrades to multiple target devices);

converting, at the device, the user-defined task sequence to a set of one or more steps of a job to be carried out to automatically deploy an operating system to the multiple additional devices (col. 7, lines 27-45, col. 11, line 58 to col. 13, line 15, col. 1, line 64-65, the parameters will be used to select the sections of the upgrade that need to be performed which are the job tree being performed by upgrade, the upgrade process is automatically implemented)**, the set of one or more steps comprising**:

configuring firmware of the multiple additional devices (col. 2, lines 23-28, col. 2, line 66 to col. 3, line 6, upgrade the operating system, firmware, applications and data files);

downloading an operating system to the multiple additional devices (col. 2, lines 16-18, install a new version of the operating system) but does not explicitly disclose **by copying an operating system image file to the multiple additional devices**.

However, Paul discloses the operating system downloading is performed by copying an operating system image file to the multiple additional devices (Figs. 1A, 4, Para. 0031, 0039, 110-114 also are connected to network 102. Clients 110-114 can employ boot servers 116 and 118 to receive operating system files and/or application files, the client PXE support to request and install additional files via M/TFTP from the boot server containing executable images of an operating system, appropriate communications and other device drivers, and other system software).

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Denby with the teachings Paul to dynamically direct clients to boot servers that have relatively low demands in order to provide the best possible service to the requesting clients (Para. 0017).

Denby in view of Paul further discloses:

rebooting the multiple additional devices (col. 7, lines 14-18, 24-25, The rebooting and process control is done automatically); **and configuring the operating system of the multiple additional devices** (col. 2, lines 8-9, col. 3, lines 33-35, col. 4, lines 10-16, col. 14, line 60 to col. 15, line 3, col. 19, lines 37-39, A new config.sys and startup.cmd will be copied over the old so that when the DCS 300 reboots); **and**

sending one or more commands configured to carry out the one or more steps of the job (col. 7, lines 24-59, col. 11, line 25-26, col. 17, lines 46-56, start upgrade from the GUI by calling upgrade utility) but does not explicitly disclose **wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices**.

However, Devine discloses wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices (Fig. 6, clients connected to server, Fig. 10, clients receive data service, col. 9, lines 1-17 and 61-67, clients exchange data with servers and software upgrades are performed by distributed administrator, a server, col. 46, lines 41-49 distributed administrator provides both synchronous and asynchronous services, col. 47, lines 27-31, service is delivered synchronously or asynchronously to clients, the connected devices).

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Denby with the teachings Devine and Paul to reduce downtime and to free

developer from firmware upgrade as it has been realized that the asynchronously and synchronously software upgrade is efficient and cost-effective (col. 52, lines 39-65).

As to claims 24 and 26, have the same subject matter as of claims 3 and 8 respectively and are rejected under the same rationale, based on teachings from Denby.

Claims 5 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Denby in view of Devine, as applied to claims 1 and 13 above, and further in view of Hsieh et al. (US Patent Publication 2002/0191014)

As to claim **5**, Denby in view of Devine discloses the method as recited in claim 1, but does not explicitly disclose wherein the task sequence is part of an Extensible Markup Language (XML) file.

Hsieh discloses the task sequence is part of an Extensible Markup Language (XML) file (paragraph 0044, the user interface 40 communicates with the gateway 38, which converts messages into the appropriate format. For instance, the gateway can convert SQL data messages from the database 32 into an XML (Extensible Markup Language) format which the user interface 40 then processes into a presentation format for display to the user).

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Denby in view of Devine with the teachings Hsieh to communicate with

device using XML format to install the upgrade across a wide range of product characteristics (Denby, col. 1, lines 41-46).

As to claim 14, is directed to a computer readable medium carrying instructions for performing the methods of claim 5 therefore rejected along the same rationale.

Claims 39-43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Denby et al. (US Patent 6,976,062, hereinafter Denby) in view of Devine et al. (US Patent 6,944,662, hereinafter Devine), and further in view of Pierre-Louis et al. (US Patent Publication 6,421,777, hereinafter Pierre-Louis)

As to claim 39, Denby discloses **a system** (Fig. 1) **comprising**:

a controller, stored on one or more computer-readable storage media and configured to be implemented at least in part by at least one of one or more processors to obtain a task sequence that describes one or more steps to be performed on multiple remote devices (Fig. 1, 101, upgrade utility [controller], col. 4, lines 42-61, the upgrade utility 101 examines the ADC device platform 105 for its characteristics pertinent to the upgrade. Alternatively, the upgrade utility 101 may consult the database 106 for pertinent ADC device platform characteristics, col. 7, lines 14-34, The parameter list in the upgrade.ini file controls the upgrade behavior, col. 11, line 58 to col. 13, line 15, upgrade.ini file typically contains tags and parameter definitions, the parameters will be used to select the sections of the upgrade that need to be performed, col. 2, lines 8-9, initiate simultaneous upgrades to multiple target devices), **and to generate a job representation of the one**

or more steps (col. 7, lines 27-45, col. 11, line 58 to col. 13, line 15, the parameters will be used to select the sections of the upgrade that need to be performed, which are the job being performed by upgrade), **the one or more steps** configured to perform **at least one of:**

configuring firmware of the multiple remote devices (col. 2, lines 23-28, col. 2, line 66 to col. 3, line 6, upgrade the operating system, firmware, applications and data files);

downloading an operating system to the multiple remote devices (col. 2, lines 16-18, install a new version of the operating system S);

rebooting the multiple remote devices (col. 7, lines 14-18, 24-25, The rebooting and process control is done automatically); **or**

configuring the operating system of the multiple remote devices (col. 2, lines 8-9, col. 3, lines 33-35, col. 4, lines 10-16, col. 14, line 60 to col. 15, line 3, col. 19, lines 37-39, A new config.sys and startup.cmd will be copied over the old so that when the DCS 300 reboots); **and**

a network boot service, configured to be implemented at least in part by at least one of the one or more processors to detect when the multiple remote devices are coupled to a network that the system is also coupled to, and to communicate with the controller to determine which of the steps of the job representation are to be carried out in response to the detection (Fig. 1, col. 4, lines 17-61, upgrade utility and ADC devices are communicated through network, col. 4, lines 42-61, the upgrade utility 101 examines the ADC device platform 105 for its characteristics pertinent to the upgrade), but does not explicitly disclose **wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more**

steps concurrently for the multiple additional devices, but does not explicitly discloses a network boot service to detect the multiple remote devices.

However, Devine discloses "wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices" (Fig. 6, clients connected to server, Fig. 10, clients receive data service, col. 9, lines 1-17 and 61-67, clients exchange data with servers and software upgrades are performed by distributed administrator, a server, col. 46, lines 41-49 distributed administrator provides both synchronous and asynchronous services, col. 47, lines 27-31, service is delivered synchronously or asynchronously to clients, the connected devices).

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Denby with the teachings Devine to reduce downtime and to free developer from firmware upgrade as it has been realized that the asynchronously and synchronously software upgrade is efficient and cost-effective (col. 52, lines 39-65).

Denby in view of Devine does not explicitly disclose a network boot service to detect the multiple remote devices.

Pierre-Louis discloses a network boot service to detect the multiple remote devices (Figs. 1, 4, item 104, col. 3, lines 3-5, server 104 provides data, such as boot files, operating system images, and applications to clients 108-112, col. 5, lines 9-15, track [detect] the remote system's reboots, determine the state of the client computer, determine the appropriate boot image for the current state, switch the boot image when necessary so that right boot image is downloaded to the remote client system at the next boot request).

Therefore, it would have been obvious to one skilled in the art at the time of the present invention to modify the method of Denby in view of Devine to include remote boot services as taught by Pierre-Louis to effectively manage and upgrade remote devices connected to a network (col. 1, lines 43-64).

As to claim 40, Denby in view of Devine and Pierre-Louis discloses the system as recited in claim 39, wherein the one or more steps includes steps for automatically deploying an operating system on the remote device (col. 2, lines 16-18).

As to claim 41, Denby in view of Devine and Pierre-Louis discloses the system as recited in claim 39, wherein one of the steps comprises another task sequence (col. 20, lines 48-52, reboot step comprises sequence of saving and rebooting system).

As to claim 42, Denby in view of Devine and Pierre-Louis discloses the system as recited in claim 39, wherein one of the steps comprises an operation to be performed on the remote device (col. 2, lines 14-15, Allow for upgrades to occur from remote, centralized locations).

As to claim 43, Denby in view of Devine and Pierre-Louis discloses the system as recited in claim 39, wherein the job representation comprises a tree having a plurality of nodes, and wherein each of the one or more steps is represented by one of the plurality of nodes (col. 7, lines 27-34, is a tree of sequential step nodes for upgrade).

Response to Amendment and Remarks

Applicant's arguments based on newly amended features, such as "wherein the one or more commands are configured to carry out at least one of the one or more steps asynchronously for the multiple additional devices, and are configured to carry out at least one of the one or more steps concurrently for the multiple additional devices", "by copying an operating system image file to the multiple additional devices", " have been fully and carefully considered but are moot in view of the new ground(s) of rejection. Refer to the corresponding sections of the claim analysis for details.

At Page 13, concerning 35 U.S.C. § 101 rejections made against claims 39-43, Examiner respectfully submits that amendments made to the claims does not provide physical structured components rendering the claims in the category of machine under 35 U.S.C. § 101, including the amendment made to the controller – "stored on one or more computer-readable storage media". The claim as it is written, the storage media does not appear to be part of the system. The Examiner suggests amending the claim to explicitly include the storage media as part of the system.

At Pages 17-21, with respect to claims 22, 27 and 36, concerning Applicant's arguments about the newly amended subject matter of steps carried out "to automatically deploy on operating system to multiple devices", Examiner respectfully submits that it has been held that broadly providing an automatic or mechanical means to replace a manual activity which accomplished the same result is not sufficient to distinguish over the prior art. See MPEP 2144.04 section III, *In re Venner*, 262 F.2d 91, 95,120 USPQ 193, 194 (CCPA 1958).

Conclusion

Applicant's amendment necessitated the new grounds of rejection presented in this Office Action. Accordingly, **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shew-Fen Lin whose telephone number is 571-272-2672. The examiner can normally be reached on 8:30AM - 5:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hosain Alam can be reached on 571-272-3978. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Shew-Fen Lin /S. L./
Examiner, Art Unit 2166
January 5, 2010

/Hosain T Alam/

Supervisory Patent Examiner, Art Unit 2166